$7.00 U.S.

INSIDE: E-mail From MultiValue

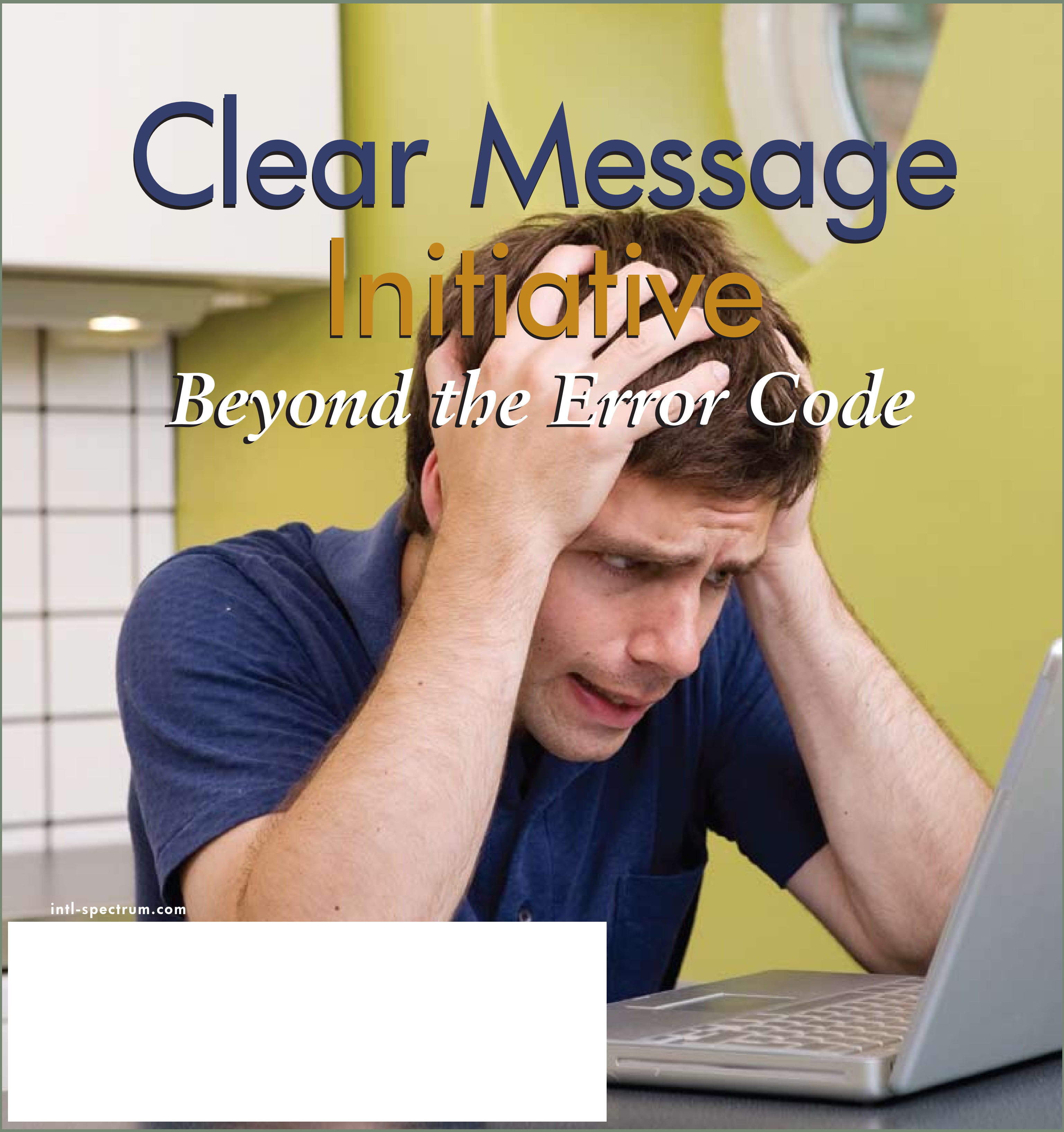
INTERNATIONAL

Spectrum®

THE MULTIVALUE TECHNOLOGY MAGAZINE | JAN/FEB 2011

PLUS!

Using QuickBooks With MultiValue



Clear Message Initiative

Beyond the Error Code

intl-spectrum.com

INTERNATIONAL Spectrum®
THE MULTIVALUE TECHNOLOGY MAGAZINE

FEATURES | JANUARY/FEBRUARY 2011

DEPARTMENTS

From the Inside

BY NATHAN RECTOR

New Year, New Features, New Resources, New Formats...

We have a lot of new things and changes happening this year; one is the magazine. One of the big pushes this year is to make the magazine more digital friendly. As a start, you'll see QR Codes on each article.

If you do not know what QR codes are, then here is a quick summary. QR Codes are bar codes that are readable by cell phone cameras. These bar codes contain information like urls, text, sms messages, e-mails, vcards, and much more. In our case, the QR Codes contain URLs to the magazine articles that are mobile device friendly.

We have had a digital edition of the magazine for several years, and we have a large archive of digital magazines, but they are sometimes hard to search and don't re-link related content. The files sizes for the PDFs are a little unrulely for mobile devices, so we are introducing new formats.

We aren't just making the magazine more digital friendly, but we will be recasting much of the resources and information available from Spectrum in a like manner.

Kindle, other eBook formats, iPhone, Android, iPad, e-mail, Web, social media... all of these formats will begin to work together as extensions of each other to provide you, as a MultiValue developer and user, the resources, tips, and education you need.

In this new digital format, you will start seeing a related links section. The related links will help provide you more information, and other resources you might find of interest based on the content and references in the current article you are reading.

As part of the change in the digital resources, the e-mail newsletter is also going to be extended this year to provide you more information about the products and tools you use. You will be able to customize it to include the information you want and what is relevant to you.

Also look for mobile and social apps that combine all the features of the web site, magazine, and newsletter into one place.

In summary, 2011 will be a year of digital media.

If you are interested in becoming more widely recognized as an expert in the MultiValue market, now is the time to start submitting articles, case studies, and tech tips to us. If you want to find out more about the programs we have for authors and consultants, then feel free to contact us (nathan@intl-spectrum.com or editor@intl-spectrum.com).

-NATHAN RECTOR
President, International Spectrum
nathan@intl-spectrum.com

twitter.com/intlspectrum   intl-spectrum.com/facebook

INTERNATIONAL Spectrum
JANUARY/FEBRUARY 2011

NATHAN RECTOR
President

CLIFTON OLIVER
Editor

TRACEY RECTOR
Layout

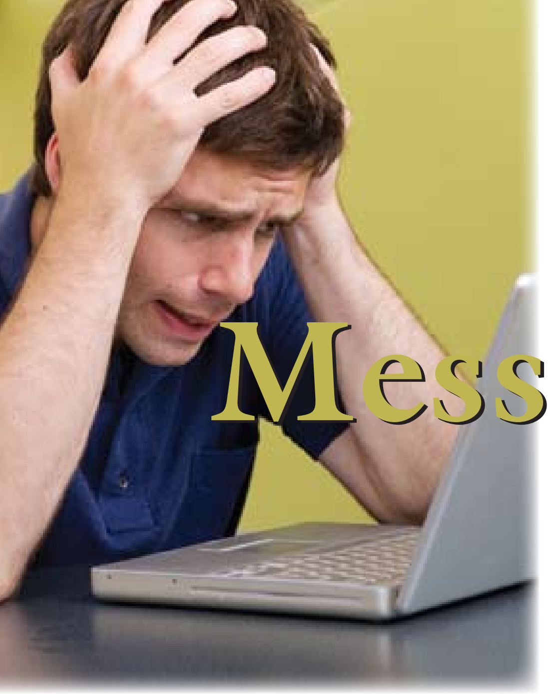
Learn more about the MultiValue Symbol and see what MulitValue Technologies and MultiValue Communities exist to help you support and manage your business and systems. To find out more visit
http://www.intl-spectrum.com

MISSION STATEMENT *International Spectrum* magazine's editorial mission is to be the premier independent source of useful information for users, developers, and resellers of MultiValue database management systems, open systems business database solutions, and related hardware, software, and peripherals. Published bimonthly, International Spectrum provides comprehensive coverage of the products, companies, and trends that shape the MultiValue marketplace as well as the computer industry at large — helping its readers get the most out of their business computer systems.

International Spectrum is published six (6) times per year at the subscription price of $40.00 U.S. in the U.S.A.; $45.00 U.S. in Canada and Mexico; $50.00 U.S. for other countries. Single copy rates are $7.00 U.S. in the U.S.A. and Canada, and $9.00 U.S. in all other countries. International Spectrum is published by International Specturm, Inc., 8956 Fox Drive #102, Thornton, CO 80260; Tel: 720/259-1356; Fax: 603/250-0664 E-Mail: request@intl-spectrum.com.

PRINTED IN USA

NEWS RELEASES/UNSOLICITED ARTICLES
International Spectrum is eager to print your submissions of up-to- the-minute news and feature stories complementary to the MultiValue marketplace. Black and white or color photographs and diagrams are welcome. Although there is no guarantee a submitted article will be published, every article will be considered. Please send your press releases, articles, and queries to: editor@intl-spectrum.com. International Spectrum retains all reprint rights.



Clear Message Initiative

BY SUSAN JOSLYN

"Danger! Danger!"

This warning was issued frequently from the tinny speaker box of the unnamed robot in the 1960's television series *Lost in Space*. The message communicates to a limited degree. It does successfully identify the general (dangerous) condition. It doesn't offer any information about the danger or advice about what to do, but the fact that it is a warning about danger is clear enough. It gets a bit better when specifically directed: "Danger, Will Robinson!" Still no indication of what is dangerous or what our intrepid Will Robinson should do about it. But it is more targeted and specific; it is Will Robinson who is in danger. Another oft-quoted warning message from these earliest — albeit science fiction — days of computing was, "Does not compute." This one gets top honors for useless information. It just doesn't get any more vague than that! As the years have passed, there has been much progress in technology, but not as much progress in message clarity. Bad error messages may, in fact, be the bane of the modern world.

Recently I was unexpectedly thwarted in the routine task of syncing my smartphone. An option-less dialog announced, "Function OpenFolder failed." Complete fail. I can venture a guess or two about the cause, using my own programmer expertise, honed deductive reasoning skills, and knowledge of the English language. Some folder has gone missing or corrupted. I wish I knew what might have changed. Or maybe if the message had indicated the name of the folder, or what the program was trying to do, more broadly when it encountered the problem, these clues would have given me something to try before that age-old catch-all solution — delete and reinstall the whole application. How much time do we waste on that particular activity?

Not only is my life filled with cryptic, inexplicable, counter-productive error messages, but I have come to realize that my own software is positively riddled with them! Until recently I guess I figured that getting the software to work was important and reporting when it didn't was not. Nobody reads these messages anyway, was my thinking. They'll call (or e-mail) me. And they do. But this is a self-perpetuating cycle! They might not have to turn to me if the error messages and warning they received made clear sense and offered an alternative.

To be fair to our ancestors, resources were different in the breaking dawn of the information age. Each letter-character required time, effort, and money to display (or print). Options for output were limited. Messages were necessarily cryptic. The limitations are mostly long gone, now, but the culture that had grown up around them has remained. Maybe, to our credit, programmers just like to be *efficient*. We can probably agree that a common generalization about programmer-types is that we (well, some of you) tend to use words sparingly. Or it may be that programmers have their own language and non-programmers just "don't compute." Certainly users of an application appreciate different things in an error message from what the application's designers may want to know. Many a true geek believes that the classic DOS error message BAD COMMAND OR FILE NAME is a thing of true beauty. It is elegant in its simplicity — and so flexible! It can be applied to such a wide range of circumstances — one size fits all! What it is not, however, is helpful.

So, here we are. You, me, your mom, your kids. Surrounded by fast moving technology that is fraught with errors and very little guidance. If we are going to progress at this rate — as a society, as a nation ,or as an industry — we have got to get better at reporting and explaining failure. And we have to offer guidance that doesn't require translation!

A successful error message may or may not include humor, but it should definitely be informative.

What do we fix in figure 1? Are we, as programmers, *lazy?* Do error messages fall into the dreaded category of "documentation," to be avoided at all costs? Do we believe that we are somehow protecting our users from knowledge that they can't possibly comprehend? Or are we protecting something proprietary for ourselves? While it is true that keeping things mysterious will help us retain our all-knowing mystique, it will also keep us tied to the phone answering the questions that the error message could have answered.

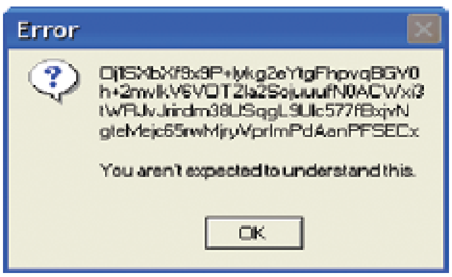


Fig. 1

When my own error messages are lame, the most likely reason is because I really do have this notion that no one will really *need* them. I have to put something there; it's just what you do (or there is a mandatory ELSE clause), but I don't really expect anyone to have to rely on the message. Like life insurance — you buy it, but you can't really think about anyone ever *really* needing it. Yes, I do know that is crazy. And it's not a conscious belief. But I think it governs my behavior to some extent, and I don't think I'm alone in this.

Blogger Harry McCracken theorizes in his *Technologizer* entry The Thirteen Greatest Error Messages of All Time, that error messages should be painful at worst and boring at best. He contends that, "They tend to be cryptic; they rarely offer an apology even when one is due; they like to provide useless information like hexadecimal numbers and to withhold facts that would be useful, like plain-English explanations of how to right what went wrong. In multiple ways, most of them represent technology at its most irritating. They're rarely helpful. Actually, they usually add insult to injury. But what would computing be without 'em?"

We can joke about it. We can wax nostalgic. An error message from a machine out of your past can transport you back in time. But what we need to do — as a community, as an industry, and as right-thinking humans — is make error messages and warnings better! If we want to provide truly great software — and we do! — but we can't make it flawless — and we can't! — we must make it fail gracefully. We must offer a path to usability even, especially, in the event of a failure. Thus, I contend, the secret to quality software and customer satisfaction is communicating clearly through error and warning messages.

Let's start a revolution! Are you with me?

Take a look at the classic example — the http (404) "File not found" error (fig. 2). It is inarguably the most common error message in the human experience. In its early incarnations it was fairly unhelpful (see figure). Grating. Confusing. Negative. Bam! Denied!

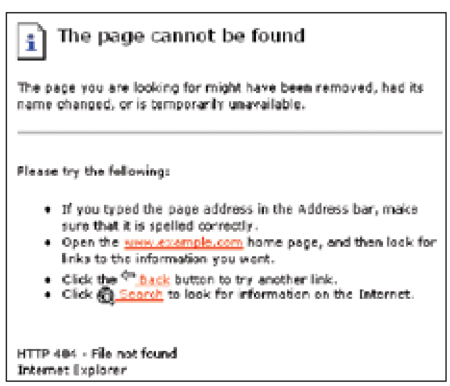


Fig. 2

But it was customizable and some enterprising organizations took it further. In fact, if you would like to have a laugh, use Google Images for the phrase '404 File not found' and look at what people have done with the place! (Or go to errorware.com where you can buy clothing with this and other great error messages emblazoned on it!)

So, many web sites offered more polite versions. Some offered more information, included a graphic, a joke, or even a joke graphic. Those organizations have recognized a universal truth that is too often overlooked: when people hit an error and are prevented from doing what they meant to do, they will have an emotional response (and you know the emotion I mean). Throwing in something fun to make the user smile through their pain is not a bad idea, even if it doesn't help the user solve whatever problem they are facing.

A successful error message may or may not include humor, but it should definitely cover these points information:

- The severity: Whether this error critical, fatal, a warning, or just a reminder.
- The problem: what has occurred, what should have occurred, the reasons, and the impact.
- The application layer: what application (or layer) has produced the error.
- The action: What you can do about it right now, how you can find out more, and how you can prevent reoccurrence.

The challenge is to convey all of this information in a succinct 80 to 200 characters. It isn't impossible. Look what we're doing on Twitter with 140 characters.

```
87yr ago natl dads: "All
equal." War tests. Battle
hallowed ground > our words.
We vow: dead not in vain,
govt of/by/for peeps here 4
keeps
```

Yes, that was Abraham Lincoln's Gettysburg addressed boiled down to a 140 character tweet. Twitter — and text messaging — have inspired a whole

Continues on page 8

pseudo written language (as well as the dismaying notion of tiny URLS, but that's a topic for another day). These types of shortcuts would rarely be acceptable in a software application's error messages. So, we have to find a way to convey the information briefly but in more formal language, at least for now. This is probably best achieved by using a two

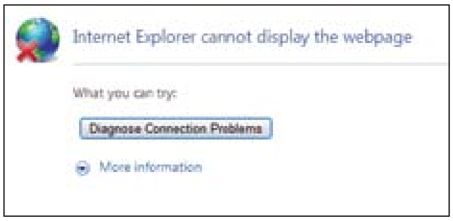


Fig. 3

part message. The main message text is like a heading. It must be concise, definitive, and structured. It should be instantly recognizable. Then a link, button, or second strike of the function key can take the user to a more elaborate explanation. Going back to that classic "404 File not found" error. Figure 3 shows how it has evolved in Microsoft's Internet Explorer 8.

Aside from the questionable graphic where the user is denied the entire earth, this is clear enough and easy to recognize by time you've seen it a few times. It is very general, but its saving grace is the magic "more information" arrow option. That second layer of help is where we can really communicate. Figure 4 is what you see when you click on that.

This addresses all of the elements, but it has so many maybes that it almost cancels itself out, and it is far too many options for a casual user to absorb. In the time it took to think up and type in all those maybes, at least in our easy-string-manipulation culture, we could have figured it out a bit and at least limited the choices. This is where we (MultiValue/U2 programmers) can use the flexibility of our platform to separate ourselves from the herd. But I digress.

This message is better . But what makes an error message good? Yes, it should have all the pieces of information. But because that information is crammed into a small place and not really going to be read in depth, *how* it is organized and displayed is going to matter just as much as what it says. These are some starting points that can get the conversation started.

Standardize

There aren't any unified guidelines for error messages. Some application development environments have their own graphics and character-length limits. Microsoft has some general standards. Some standardization with other commonly used applications is helpful to the masses. But what is more important is that a standard be applied within the application.

Recognizable

If there is a main word or phrase that is commonly used it can be instantly recognized and absorbed. "Not on file" or "Required fields missing." Whatever the specifics — and there should be specifics! — that come after, these broader category-phrases will help. Using graphics to standardize the severity or layer of error is a very nice touch if that is available in the platform. Anything that makes it quicker to understand for our human users who understandably have very little patience for error messages.

Layer

We work in complex environments. When something fails, it could be something in our application, something in the application development environment, something in the underlying operating system, something with the network, or even something with the server. That could be the host, the client, or some server in the middle. It is a great help to identify the *layer* where a problem has occurred in the error message.

Syntax, grammar and punctuation and spelling

For the linguistic sticklers among us — and we know who we are — poor syntax, grammar, or punctuation can feel like the screech of chalk on blackboards. It is simple enough to check a style guide if you are not sure. A small typo can end up preserved like a fossil for all posterity.

Word order, tense, passive vs active

Passive? Nouns first? "Customer name must be alphanumeric." Active, lead with a verb? "Enter alphanumeric customer name." Or passive with the adjective descriptor first? "Alphanumeric is required on customer name field." "Decimals are not allowed in quantity field." Whatever you prefer, if you decide on it and stick with it, uniform error messages will be more quickly parsed and absorbed by the users.

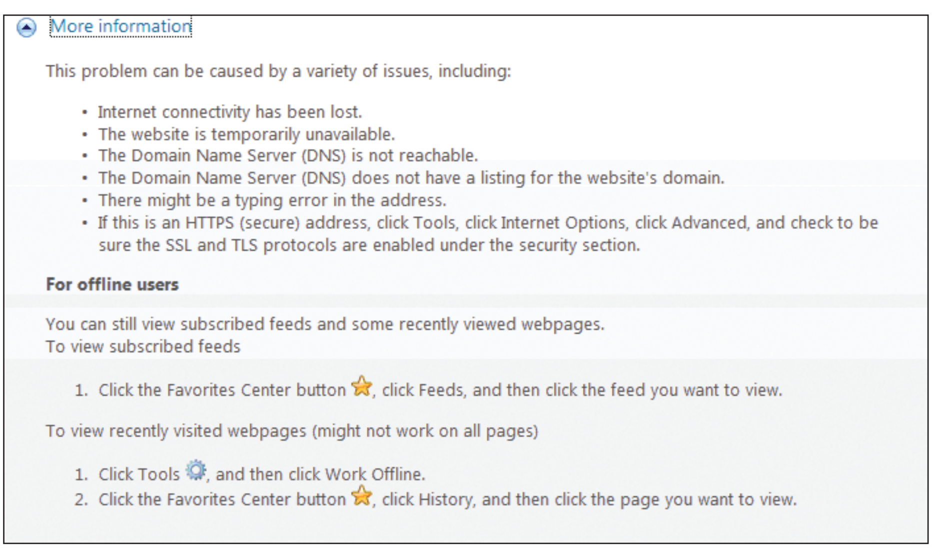


Fig.4

If you use active/directive commands, be careful with the tone. According to some guidelines it is better to avoid accusative or admonishing tones. As rewarding as it might be to say, "You entered a bad name!" you can almost hear "jerk" at the end. Or, "Please don't try to use an alpha character in a date field." Again you can almost hear the exasperated again at the end. Expressing our own frustration as programmers only fosters bad feelings.

Specificity vs Generalization

Reusability is a great concept for development. But when it comes to errors, we must be careful not to generalize so much that our messages become useless. Many applications have been designed with error message databases that can be referenced by number. This can be a boon to standardization and a great way to provide lots more information for that second click. But we run the risk of starting to think of those error messages as precious resources again. Perhaps it feels sloppy to have a lot of similar messages. There are ways to combat the sloppy feeling and address the inefficiencies while still providing useful, specific information (variables in the error message, mainly). Or let go — let it be inefficient underneath so that it can be more useful on the surface! Messages need to offer specific information to the user. What may feel elegant and streamlined to the programmer is not always the most satisfying user experience. Take an extra moment when crafting an error message to see how much information you can give the user about the specific failure in terms that are meaningful to what the user will be trying to accomplish.

Let's start a revolution! Let's look like the best applications out there even when our interface isn't the flashiest graphic. Because even in a green-screen environment, we have the powerful and easy programming language to allow us to do that. If we remember what it is like to be that poor befuddled user trying to sync our smartphone, and if we assume that our error messages will be seen, that they will be needed, we can do better. Above all, let's remember that our error messages are a reflection of ourselves and of the quality of our workmanship!

Linked IN: Want to contribute ideas and suggestions – perhaps gain insight about what others are doing? Join us in the newly created Linked In group "Clear Message Initiative". IS

SUSAN JOSLYN is the President of SJ+ Systems Associates, Inc. and is the author of PRC?, a complete, integrated software development life-cycle management / IT Governance tool for U2. She has worked with U2 (nee Pick/Multivalue) and SB+ software the beginning (both hers and its) and has specialized in IT Governance, including quality, compliance and life-cycle productivity issues since the early 1990's.

E-Commerce Made E-Z

BY BRYAN SHUMSKY, REVELATION SOFTWARE

MultiValue applications fill many niches, but most of them are ultimately involved in some way with the buying or selling of goods or services. In the pre-Internet and early-Internet days, collecting or sending the funds to finalize these financial transactions was done manually, "off line" and outside of the application. This may have originally involved wire transfers or checks, though many businesses eventually added support for credit card payments, either via manual card readers or various third party payment processors.

In today's Internet-connected world, however, as our applications are migrating from the green-screen desktop to the rich browser on the web, the need for customized payment transaction software has been reduced or eliminated, as several large vendors have provided open connectivity that any web application can use. Google and Amazon are two of the big names, but the five-hundred-pound gorilla of payment processing has to be PayPal. PayPal was the strongest of several financial transaction companies in the early 2000s, but it really rose to preeminence after it was acquired by eBay. Today, PayPal handles over 15% of US e-commerce transactions, and nearly 10% of e-commerce transactions world-wide.

Over a year ago, PayPal opened up its APIs so that other software could take advantage of its peer-to-peer network.

If the "bottom line" in your business is to keep a healthy bottom line, the ability to get paid quickly and easily has to be a top priority.

While several sets of APIs allow the payment transactions to be handled "behind the scenes" by PayPal, while leaving the user interface to the individual application, another set — the Website Payments Standard — is the simplest solution to use. By providing a set of simple, standardized calls, any application can now offload the details of the transaction, and users (many of whom may already have PayPal accounts) can interact with a standard, familiar payment interface.

In order to accept Paypal payments, you must sign up for a "business" account; go to http://www.paypal.com for more information. Note that once your business account is established, you can specify a number of convenient defaults for order processing (including whether your customers must be Paypal customers themselves, or whether the processing should be "Paypal optional"; whether you want shipping costs calculated automatically, or passed in with each transaction; and many others).

Once you have established your Paypal business account, you will be issued a Paypal business ID; you can use either this value, or the email address you specified for your Paypal account, when interacting with the Website Payments Standard.

Website Payments Standard includes three types of interaction: Buy It Now, Add To Cart, and View Cart. Buy It Now is used to add a single item to PayPal's "shopping cart" and begin the checkout process; Add To Cart can add a single product or service to the shopping cart, but then allows you to continue shopping; and View Cart displays the current items in the cart and can start the checkout process. PayPal can make these actions available even to non-programmers, letting you create custom buttons on the PayPal website that embed many of the required details, and which you can then place directly onto your web page. Fundamentally, though, each of these interactions is controlled by sending https:// web requests to PayPal, along with "name/value pairs" defining the details of the action, and it is by using these web requests that you have the most flexibility.

In OpenInsight's web development tool, O4W, the details of invoking the PayPal functions have been hidden behind O4W's own O4WPayPal routine, but developers in other environments can also code these directly. You must create an HTML form, or use an HTML

```
<form enctype="multipart/form-data" method="post" action="https://www.paypal.com/
            → cgi-bin/webscr" name="o4wpostform">
   <input name="cmd" value="_xclick" type="hidden">
   <input name="business" value="paypalmgr@revelation.com" type="hidden">
   <input name="currency_code" value="USD" type="hidden">
   <input name="lc" value="US" type="hidden">
   <input name="item_name" value="sample buy it now item" type="hidden">
   <input name="amount" value="10.99" type="hidden">
   <input name="item_number" value="sampbin001" type="hidden">
   <input name="button_subtype" value="products" type="hidden">
   <input name="bn" value="PP-BuyNowBF:btn_buynowCC_SM.gif:NonHosted" type="hidden">
   <input name="return"
value="http://www.revelation.com/o4w/oecgi3.exe/O4W_ORDERFORM?O4WEvent=Paypal&amp;
            → O4WControlBackup=buyitrtn" type="hidden">
</form>
```

Fig. 1

IFRAME, to submit your request to https://www.paypal.com/cgi-bin/webscr, passing in the desired command (via the "cmd" variable) of "_cart" (for Add To Cart or View Cart), or "_xclick" (for Buy It Now).

The other name/value pairs that are required for Buy It Now processing include "business" (your PayPal ID), "currency_code" (for US processing, "USD"), "lc" (locale, "US" for US vendors), "item_name" (the name of the item to show up on the PayPal description), "amount" (the amount for the product or service), "item_number" (your product number), and "button_subtype" ("product" for a product, "service" for a service). You might also provide "quantity" (if not specified, defaults to "1"), "shipping" (any shipping cost), "tax_rate" (the tax percentage), and various demographic fields if you want them pre-filled on the PayPal checkout. You may also specify the URL you would like PayPal to return to after the transaction has been completed, via the "shopping_url" variable.

For example, the O4W PayPal request:

```
O4WPaypal("BUY", "buyitrtn",
"paypalmgr@revelation.com",
"sample buy it now item",
"sampbin001", "", "10.99")
```

would generate the HTML in figure 1 and, when clicked, would generate the action in figure 2.

The "Add To Cart" action supports the same name/value pairs as discussed above; the O4W PayPal request:

Continues on page 23

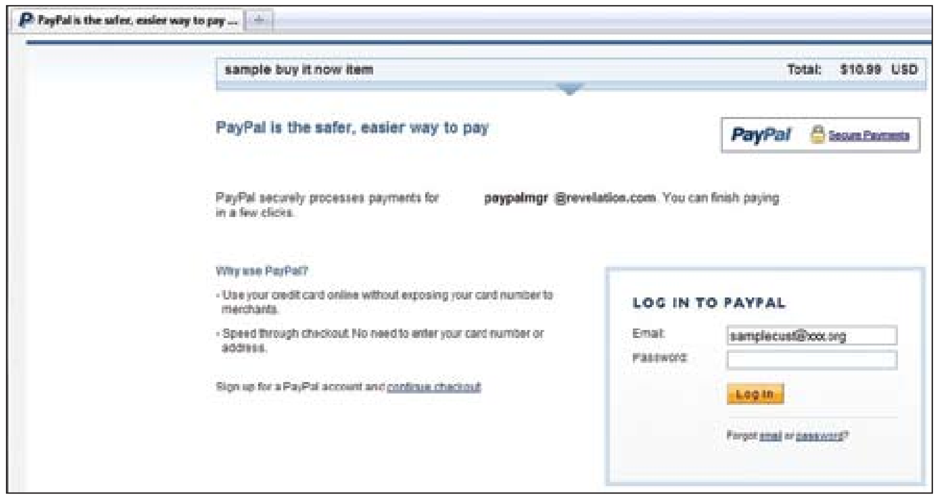


Fig. 2

Money

BY CHARLES BAROUCH

Brother Can You Spare a Dime

The economy is recovering. That's what I hear, anyway. I know enough unemployed people to make me sure that the definition of recovery has changed. It isn't the only definition which is under revision. So, here are some things IT professionals need to know about money...

As we all know, internal meetings cost money because they require salaried — and sometimes consulting — butts to be in seats where they are not carrying out their primary roles. The assumed justification for any meeting is that the planning or discussion will save time at some other point, perhaps by avoiding duplicate efforts, or cross-purpose efforts. So, when management requires meeting before pre-budgeted money is spent on what it is budgeted for, it means that management feels that the lost productivity is warranted by the fear of overspending or misspending. More than ever before, your boss is under scrutiny when he spends.

For a comparatively few companies, this is not new. For most companies, it is a complete reversal of existing policy. So when we want to attend Spectrum, or other education events, we have more hurdles to leap. Likewise, if we want to provision a new server or add a new laptop, we are likely to see unusual levels of scrutiny. That's how it affects us personally. Let's talk about how it changes our job definition.

Money For Nothing

New income is not the only way to stretch a budget. Found money — money that is being spent inefficiently or for unneeded services — is another way for companies to stay economically competitive. For IT professionals, that means that Business Intelligence, asset audits, and monitoring systems are "money finding tools" for many businesses. Additionally, anti-theft measures, like RFID-ing materials and setting readers by every exit point, are as good as gold. There are a lot of IT projects which can more than pay for themselves in found money.

We need to bring these sort of topics forward to the upper ranks. Too many people think of IT as an expense department, but we provide value for that expense and, with projects like these, we are a margin-retention department. When people look at the Sales department, for example, there are certainly expenses. The offset is the value demonstration that Sales can produce — a list of orders. We need to think the same way in IT. What is our value demonstration?

I worked in one company where IT found significant mismanagement of finances by one executive. Another executive was called to task when IT proved that he had a staffer who was actively doing non-work Internet access for an average of forty hours a week. IT can make a difference in both productivity monitoring and fraud detection. That's just two of the ways we can demonstrate value.

Pennies From Heaven

When I look at IT, I see a service business. Even when we are all staff and all of the work we do is internal. IT is generally run, like Payroll and HR for example, as a service business. This makes the transition from internal provider to external provider fairly smooth. As I like to point out during some of my training sessions, if you have data which has value to your customers, you have a possible professional service to offer. Say you are a shipping company and you do all the deliveries for a bank when it sends materials to its own branches. Now imagine if your company offered detailed records of all material transfers to the bank? You just became a third party inventory management company. Add a model usage program, and you can automate their re-ordering process. In two steps, IT moved the company from shipping specialists to shipping/inventory/purchasing specialists.

Even if these value-added services are sold cheaply, they bring in some pennies in a tough economy. Even if they are offered free as a form of customer retention, they provide a competitive edge and therefore, value.

Cent, Five Cent, Ten Cent, Dollar

There is more than small money at stake. When that shipping company adds enough services, it becomes a logistics company. When it expands its reach, for example by leveraging the Internet, it becomes a global logistics company. When IT facilitates these processes, we move from expanding the company's scope to transforming it completely.

To give you a non-IT analogy: In China, export quotas were divvied up amongst certain factories. Some of those factory owners found that reselling their access to quota was more profitable than running a factory. They transformed their business from a physical goods manufacturing basis to an export facilitation basis. Management is more likely to see these opportunities if they know that they exist. Letting them know what we can do isn't bragging, it's good business.

Ten Cents a Dance

We think of a tough economy as a time for belt tightening and doing without. Sometimes it is a great time to expand your use of consulting help. If a project will save the company customers, improve your relationship with a vendor, or directly increase revenue, getting it done faster might justify bringing in a hired gun. The win-win for IT is that proving the need for consulting help and using it sparingly proves you are company-focused while using it extensively becomes the basis for justifying adding staff.

When you decide on consulting help, ask the cost-benefit questions of yourself before your bosses ask them. Can I peg the consulting cost to a specific benefit? Can I show a time savings by bringing in more help? Will this stop other projects from being delayed? Will extra hands allow us to make an otherwise impossible deadline?

IT is a service business, even when we are all staff.

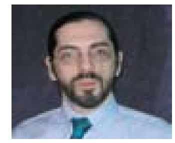
Another reason for bringing in temporary help can they can make your permanent staff smarter? If this is your justification, make sure your staff wants to learn and your consultant is good at teaching. Remember that those of us who are getting taught generally see a value. Those who pay see two-part cost: training costs plus the cost of paying the employee or employees during training. Make sure you know how this additional person will prove enough value to cover both costs.

Can't Buy Me Love

IT managers please note. If IT services are offered free to customers, insist that the value be treated as an inter-company transfer from Marketing or Sales. If they use the service you provide, they should — on paper — pay for it. Other departments do this all the time so their expenses are not exaggerated. Likewise, when you order a new desktop for Johnny in Billing, be sure the expense is recorded to Billing and not IT. If your Finance department will not support that, keep records yourself. When the owner/stockholders/bosses ask where the money went, it is handy to be able to show just how little of IT's budget was spent on IT. It is especially handy if us IT people, the folks who make reports, can accurately present details on who spent what for which project or purpose.

Money is about income and outflow. Outflow becomes an accountability issue. When we don't account for it, it becomes an "IT is costing too much issue." I don't mind getting a black eye for overspending if I'm the one overspending. It hurts twice as much when I get one for someone else's actions. It hurts triple when I could have blocked it with one ledger sheet and didn't take the time to defend myself.

While keeping better track won't buy you love, it might buy you respect. IS



CHARLES BAROUCH is the CTO for Key Ally, Inc. He is past President of the International U2 Users Group, and a regular Spectrum Magazine contributor.

Contact Charles at Results@KeyAlly.com, or phone (718) 762-3884 ext 1.

MultiValue and Quickbooks

Part 1: Importing with IIF files.

BY MARK JOHNSON

While the MultiValue database can hold its own against the databases that support contemporary looking (and feeling) applications, there are situations whereby we must yield portions of our entire application suite to more modern packaged software.

This is not a dissertation on being able to reprogram existing modules into a more modern (read: GUI) form. We all have been there and know that the cost far exceeds what the price of packaged software is.

One fortunate aspect of having a combined system (MultiValue and non-MultiValue) is that the non-MultiValue modules tend to be the lesser of the two forms of the company's systems. Meaning, if one form is production/revenue producing and the other is accounting, the production modules are the ones that have received the greater investment in improvements over the years whereas the accounting modules are simply passive and don't evolve as much.

> *Whether QB, Peachtree, One Write Plus, or Great Plains, companies kept their MultiValue production/revenue producing modules and use these packaged systems for the mundane accounting functions.*

In all my years of MultiValue consulting and programming, changes to my client's accounting modules are the least of my efforts. Once they have been installed, they tend to be good enough as they are and my efforts are spent advancing all the other modules.

In the last year, I've had two situations where my clients have converted portions of their MultiValue accounting modules into Quickbooks (QB). Whether QB, Peachtree, One Write Plus, or Great Plains (other favorites), they kept their MultiValue production/revenue producing modules and use these packaged systems for the mundane accounting functions.

QB seems to be preferred either based on price, support, ease of use, number of installations, or available pool of trained personnel. All have import functions to convert spreadsheets from other systems or manual entries.

QB has an additional file format called IIF that is very helpful for us MultiValue programmers when exporting data from the remaining MultiValue production modules into QB. IIF files are for importing into QB only.

I will use a typical MultiValue Sales Invoice Update function as an example of creating an IIF file. This assumes that the MultiValue system has an order file and a customer file. We will be exporting the Accounts Receivable (AR) portion into QB.

Basically an IIF file is a column-labeled format like a CSV with a header row but with an additional twist. Instead of the column headings being defined once on the first row, you can have multiple file

Continues on page 24

The International Spectrum 2011 MultiValue Developer and User Conference provides access to the following knowledge:

Spectrum 2011 Live is in its 30th year of providing MultiValue developers and users with support for their applications. IT demand is going up so are your requirements for supporting your business and its needs.

Spectrum 2011 Live provides you with labs and hands-on training that you cannot get anywhere else. These in-depth sessions allow you to take home knowledge that is unavailable from other sources. They also show you how you can leverage both MultiValue-specific and non-MultiValue tools and resources.

Mobile Development

Find out how to access your MultiValue data with mobile devices. As more and more people use iPhones, Androids, and BlackBerries, the need to access your data outside the enterprise is essential. If you have not developed a basic mobile system, then now is the time to create one. Studies show that you can save your company tens of thousands of dollars just mobile enabling your enterprise.

- iPhone
- Cell Phones
- Android
- WiFi enabling
- BlackBerry

Database Management

Always an important requirement for any enterprise, database management is a constant. As new systems and requirements are placed on your enterprise, database management becomes more complex, and systems that worked in the past may no longer work now

- Disaster Recovery
- Faster Backups
- Database Security

Security

Database, software, and system security is essential now more than ever. New interfaces open up vulnerabilities that did not exist in your MultiValue systems only a few years ago. New laws and mandates are imposing more complex encryption requirements and storage needs that did not exist before.

- Data Encryption
- Security Vulnerablities
- Hacking Vulnerabilities in MultiValue Databases

Web Development

Web enabling your business system has been something that IT has been working on doing for years. There are many different solutions to this, as well as providers to solve the problem. Having information available to your customers, employees, or the general public through your web site is just as important as through your desktop systems.

- Google Interfaces
- Providers and Tools
- Ajax and jQuery Solutions
- ASP.NET
- PHP

Project Management

IT managers can not effectively run their departments without project management tools and techniques. Correctly estimating the time a project will take to complete and the steps involved with completing it is essential.

- Project Estimation
- Agile Development
- Team Integration

Application Development

Application development is the key to any database or enterprise. Learn how to combine MultiValue development with .NET development. We will talk about the best way to combine the two technologies and give your .NET developers training on how to use MultiValue techniques in a language they understand.

- Architecture
- Development
- Web Development
- Mobile Devices
- Programming
- Administration

Summit Professional Education Doubles Speed & Accuracy of Order Entry with SEMtek and Informer

Challenge: Integrate Multiple Data Sources; Streamline Reporting Capabilities

Summit Professional Education (SPE), a global training company whose focus is to develop and deliver dynamic, relevant, and highly useful educational content to professional audiences, manages thousands of events annually with over 60 instructors. SPE had been using a custom-made Access database as its backend system and a commercial registration system for order management. Lack of data integration was a huge issue.

"We needed a system that handled all of our logistics as well as our customers (over 100,000) that we could trust and could provide us with a versatile reporting solution", said Jeff Watjen, CEO at Summit Professional Education. "We had to manually merge systems weekly (if not daily) and the margin of error was big".

Solution: Integrated ERP & Function-Rich Reporting Solution

"We were wary of moving to a new system that might require us to change processes and functions that are necessary for our business to succeed; implementing SEMtek and Informer not only addressed those concerns but provides us with functions and fields that have helped improve our processes and increase our business", says Watjen.

SEMtek is a complete ERP back-office solution, providing infrastructure and business functionality for many of the industry's biggest players like SPE. And with Informer, SPE has a web-based reporting solution designed specifically for ease of use by end-users to access and display information stored within their SEMtek database and any other database they use...in real time.

Results: Doubled Order Entry Speed & Improved Accuracy; Enhanced Customer Service Capabilities

SEMtek enabled SPE to eliminate unnecessary extra process steps as well as the need to double and triple check data that was otherwise in multiple locations. SEMtek has allowed SPE's customer care team to double the speed and accuracy of order entry as well as given them the ability to help solve customer service issues quickly with multiple order search options. "Informer's reporting ability is the best that I've seen out there", says Watjen. "There's virtually no report that SEMtek and Informer can't produce."

Also critical for SPE was a reporting environment that would provide information about marketing response rates and purchasing trends. "We did that analysis prior to implementing Informer but it was a much longer, manual and less accurate method", says Watjen. "SEMtek and Informer have optimized our standard reports as well as given us the ability to create more, all in less time."

Contact us for a FREE trial

For a personalized demonstration or to discuss your specific needs, call us today at 888-703-0016 or email at sales@entrinsik.com.

About Entrinsik

Since 1984, Entrinsik has been developing, implementing, and supporting database solutions.

Entrinsik's SEMtek ERP software now drives many of the nation's largest professional and continuing education profit centers. SEMtek incorporates best practices of event planning, registration processing, event fulfillment, event financial management, and direct marketing methodology into one comprehensive package for the Continuing Education, Seminar, and Conference industries.

Entrinsik's Informer Web Reporting software delivers real-time access to multiple data sources simply and easily using a built-in web-based query engine to provide a single point of secured information access for ad hoc reporting, report customization and analysis. For a product demonstration or a free 30 day trial, call 888-703-0016 or email sales@entrinsik.com. ■

Datatel Appoints Barbara Polk as Vice President, HR and Administration

Datatel, Inc. has announced that it has appointed Barbara Polk as vice president, human resources and administration. Polk comes to Datatel from XO Communications, where she was a senior vice president of human resources, and brings more than 20 years of experience in the human resources profession. In addition to the telecommunications industry, Polk has worked in the nonprofit, healthcare, energy, and retail sectors.

Before her tenure at XO Communications, Polk served at various organizations, including Reading Is Fundamental, the nation's largest nonprofit children's literacy organization, where she was vice president of human resources and administration; In2Books, a literacy-based eMentoring program that connects adult volunteers with third-, fourth-, and fifth-grade students from under-resourced communities, where she was director, human resources; and Innovative HR Solutions, a company that assists companies with improving their human resources function, where she was a consultant.

At Datatel, Polk will implement and monitor human resources programs and policies to ensure that they align with Datatel's strategic objectives and establish a positive employee– employer relationship. She will oversee employee relations, corporate learning, recruiting, compensation, benefits, administrative services, and corporate programs.

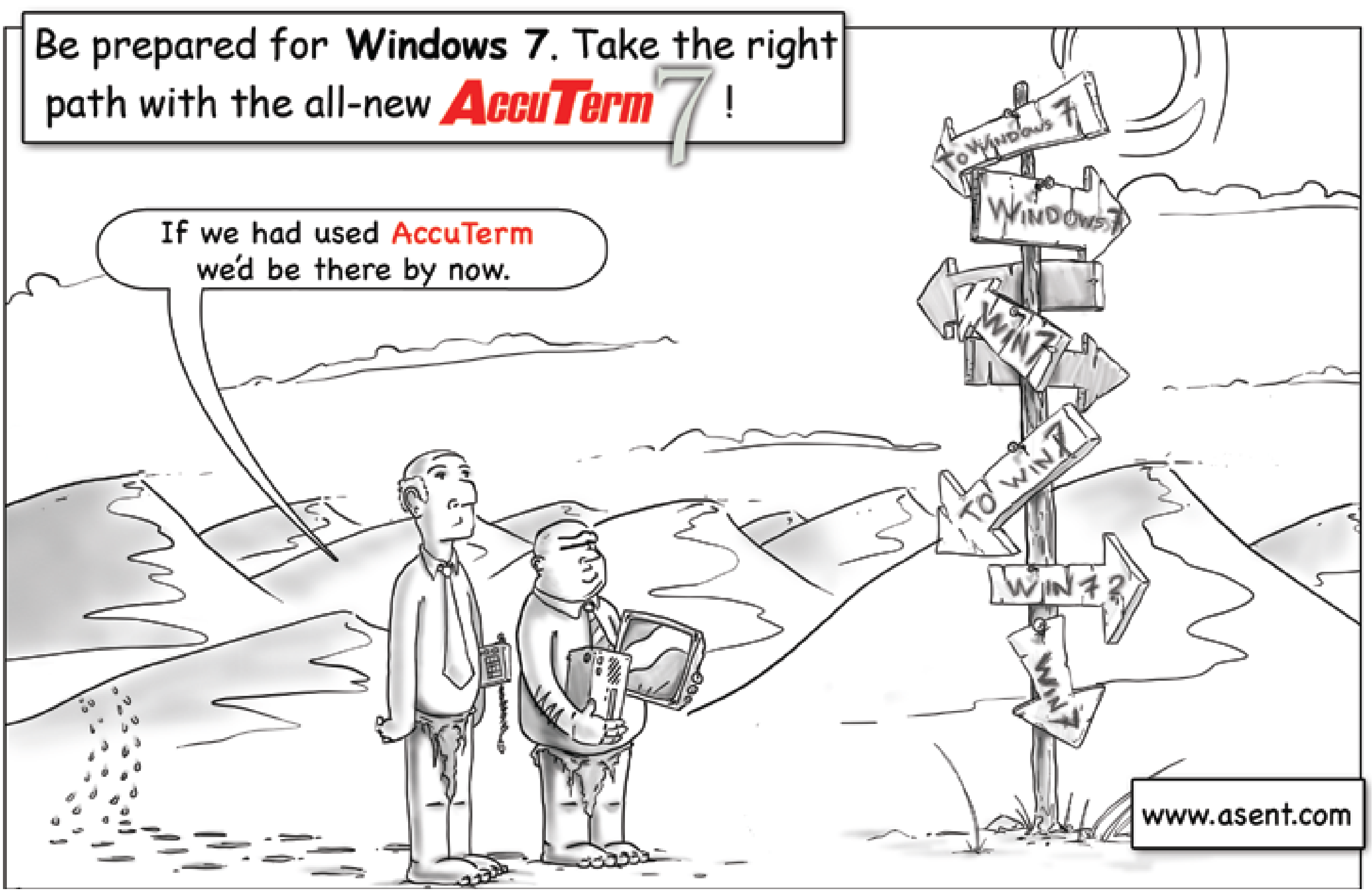
Polk holds a bachelor of science degree in political science from Rutgers University and has completed coursework for the master of arts degree in human resources management at the University of Maryland. Polk resides in Maryland.

"Barbara is an outstanding addition to our team," said Kevin M. Boyce, Datatel chief financial officer. "Her background and expertise will help us continue to build a strong, talented, and agile workforce that is able to respond quickly to the ever-changing needs of our clients and market."

"I'm delighted to join the team. Datatel's market leadership and its emphasis on its core values sets it apart from the many companies that serve the higher education market," said Polk. "I look forward to continuing to create progressive and innovative employee programs that enhance the work environment and help our clients, our employees, and our company to achieve success."

About Datatel, Inc.

Datatel is a provider of innovative technology products, services, and insight to higher education. Colleges, universities, and technical schools across North America partner with Datatel to build Strategic Academic Enterprises dedicated to achieving student success. The company has focused on higher education since 1979. For more information, visit www.datatel.com. ■



New Products

InterSystems Adds Support for Red Hat Enterprise Linux For Amazon EC2 Platform

InterSystems Corporation today announced support for the Amazon Elastic Compute Cloud (EC2) platform. InterSystems specializes in advanced integration, database, and business intelligence technologies for breakthrough applications.

Support for the latest 2010 releases of the InterSystems CACHÉ® high-performance object database and InterSystems Ensemble® rapid integration and development platform is now available on the Amazon EC2 platform with 64-bit Red Hat Enterprise Linux as the guest operating system.

"Our application partners and end-user customers have embraced cloud models and we are supporting their efforts. We recognize the growing importance of cloud-based computing," said Robert Nagle, InterSystems Vice President of Software Development. "The CACHÉ object database and Ensemble integration platform are used by customers in multiple industries to process large transaction volumes. Now, we're providing the complete support they require for high-volume processing in the cloud."

About InterSystems

InterSystems Corporation is a global software technology leader with headquarters in Cambridge, Massachusetts, and offices in 23 countries. InterSystems provides advanced software technologies for breakthrough applications. InterSystems CACHÉ® is a high performance object database that makes applications faster and more scalable. InterSystems Ensemble® is a seamless platform for integration and the development of connectable applications. InterSystems HealthShare™ is a platform that enables the fastest creation of an Electronic Health Record for regional or national health information exchange. InterSystems DeepSee™ is software that makes it possible to embed real-time business intelligence capabilities in transactional applications. For more information, visit InterSystems.com. ■

SJ+ Systems Associates Announces Cyber-Redecoration

SJ+ Systems Associates, Inc. is pleased to announce the completion of our extensive cyber-redecoration. Our web-site and e-newsletters have been updated to more modern technology, offering greater information with more convenience. Client access is easier than ever with password-protected areas for manuals, tip-sheets, presentations and white papers. We've been working hard to make it easier for our customers and prospective customers to find what they need!

PRC provides IT Governance including compliance with many regulatory agencies and initiatives. Critical documentation and transparency with a robust security framework are coupled with smooth automation, fully integrated to the U2 and MultiValue environments.

About SJ+ Systems Associates

SJ+ Systems Associates develops and supports PRC, a mature, complete Software Configuration Management Tool for U2 and Multivalue applications, including SB+. SJ+ has the experience and education in IT Governance, Audit, Software Quality and Best Practices to help customers implement a complete life-cycle management strategy that is compliant and productive. http://sjplus.com. ■

Rocket Software Announces U2 DataVu V1.1

U2 DataVu offers UniVerse and UniData users a comprehensive query, reporting and data visualization platform for desktop or web-based environments. With over 100 built-in analytical functions, U2 DataVu allows users to derive maximum value from their data and rapidly build and distribute comprehensive reporting and data visualization solutions across the enterprise. U2 DataVu allows users to:

- Develop graphical queries, visual reports and interactive dashboards
- Create reports and dashboards which aggregate data from multiple disparate data sources
- Access U2 data through native support of U2 SQL
- View data through a rich desktop application or through a Web interface
- Support a variety of output formats, including XML, HTML, Microsoft Excel(r) (CSV) and PDF
- Run on UNIX, Linux® and Microsoft® Windows® platforms

For more information on U2 DataVu please visit our Web site at: www.rocketsoftware.com/u2/products/datavu.

Free Download

U2 DataVu Query is a free download that provides fully functioning capabilities to graphically build queries against U2 Data Sources, group and aggregate to provide totals and export to PDF or print. All queries built with U2 DataVu Query can be reused in U2 DataVu Report and Dashboard without any recoding.

Visit www.rocketsoftware.com/u2/downloads/register-datavu.html to download your free copy of U2 DataVu Query. ■

New Analysis Cubes from MITS help wholesale distributors overcome complexities, boost business performance.

MITS, provider of Online Analytical Processing business intelligence and operational reporting solutions, today announced the availability of two completely new OLAP cubes for its Discover analytics solution.

Designed specifically for wholesale distribution enterprises, these OLAP cubes help business users optimize their largest asset—inventory—to increase profitability, ensure customer satisfaction and reduce costs. Users can both analyze and optimize key, inventory-related business metrics and rapidly identify both issues and opportunities when they arise.

"Inventory is a distributor's biggest investment," says Gary Owen, MITS VP of Software Services. "These new cubes help distributors look at how this asset is performing across a variety of views, each designed to highlight an area that can be improved or optimized."

The data for these cubes is sourced from popular, multivalue-based wholesale distribution management software solutions such as Activant Eclipse. Cube data content can be refreshed using a tiered-frequency strategy—with most data sourced during off-hours—to prevent any performance impact on operational distribution management systems.

Inventory Return On Assets cube boosts performance of distributors' largest asset.

Earning a high rate of return on distribution inventories is crucial. Monitoring and analyzing these returns effectively drives many of the decisions distributors must make in order to stay competitive and profitable. Regardless, says Owen, every distributor carries some inventory that doesn't perform as it should. Using critical data on inventory values, turns, sales, and gross margins, the new ROA cube lets them pinpoint inventory needing attention among products, product lines, vendors, buyers, and branches.

MITS Discover analysis cubes enable highly interactive reporting, query, drill-down and other analytic techniques. Notable reports included with the ROA cube allow users to easily identify poorly-selling but highly stocked products, top-selling but low-profit products, branches suffering decreasing inventory turns, and scores of other critically important trends.

The Inventory ROA cube—also helpful to distributors who want to reclaim warehouse space and catch slowdowns early enough to adjust accordingly—has been in use at key customer sites for several months, says Owen, and is available immediately.

Inventory On Hand cube simplifies monitoring for important trends.

Also new is the inventory on-hand cube, designed for everyday tracking of what's in stock, together with its value and status. Business users employ it to spot products with rising stock levels, for example, or to understand a product's overall supply status by matching current stock with sales commitments and quantities on order. Distributors can also use this cube to identify issues associated with defective stock, so they can take action with vendors suffering quality problems.

Fast, easy installation and administration

Owen points out that any MITS Discover solution—including those designed for wholesale distribution businesses—comes complete with software, maintenance and support, so organizations can ensure secure, high-performance data access within their own computing environments. And, because these are turnkey solutions—complete with pre-built reports—business units can quickly begin reaping the benefits.

For more details on these or any other MITS software solutions, individuals can contact a MITS representative at info@mits.com or visit the MITS Web site at www.mits.com.

About MITS

MITS is the leading developer of Advanced Reporting and Business Intelligence for the MultiValue database market and beyond. The MITS product line contains both the MITS Discover® OLAP Business Intelligence suite as well as the ad-hoc operational reporting solution, MITS Report®. Founded in 1996 and headquartered in Seattle, Washington, the company's technologies are used by thousands of organizations, resellers, and systems integrators worldwide. MITS products are firmly entrenched in a wide range of business areas, including manufacturing, distribution, retail sales and services, education, government, healthcare, and insurance as well as in many other industries. ■

Kourier Integrator for U2 - Release 3 Announced

Kore Technologies (www.koretech.com), a provider of enterprise integration and eCommerce Web solutions for MultiValue and Microsoft SQL Server databases, today announced Kourier Integrator for U2 Release 3.

Kourier Integrator for U2 is Kore's enterprise integration suite providing both Extract, Transform, and Load (ETL) and Enterprise Application Integration (EAI) capabilities for connecting Rocket Software UniData/UniVerse (U2) applications to Microsoft SQL Server and other best-in-class applications.

Continues on page 22

New Products

Continued from page 21

"U2 to SQL Made Easier" is the theme for the new release, which focuses on streamlining the process of creating a SQL data warehouse from a U2 database from start to finish while making it even faster and easier than before.

The most significant feature in this release is the new data warehouse Quick Start Workbench. This new workbench encapsulates all the tools and tasks necessary for building a Microsoft SQL Server data warehouse from U2 files into one easy-to-use interface. Users can analyze files, review data profiling information, revise file schema, and generate all of the data exports, services, and SSIS loading packages necessary to build and maintain a normalized SQL data warehouse.

"An integral part of the Quick Start tools is the ability to profile data, which allows our customers to identify and resolve potential data integrity problems in their application files," says Mark Dobransky, co-founder and managing partner at Kore Technologies. "Using the quick start and data profiling tools creates a better data warehouse resulting in more actionable business intelligence reports and dashboards."

Highlights of the new release include:

- Quick Start Workbench simplifies and accelerates SQL data warehouse development
- Unified Scheduling streamlines U2 data extraction, transformation, and SQL Server data loading into a single scheduled process
- Support for industry-standard file compression and extraction to increase data transfer performance and reduce network traffic
- Multiple options for generating SSIS packages: by File, Export, and Service Application-specific predefined SQL Server Operational Data Stores (ODS) are available for these U2 applications: Datatel Colleague, Activant Eclipse, and Activant Prelude.
- Full support for 64-bit Microsoft SQL Server and Windows Server environments
- Many user-requested and general product enhancements

For additional information regarding the Kourier Integrator for U2 solution and how it integrates MultiValue applications with Microsoft SQL Server or other databases, please contact Kore Technologies at 866-763-5673 or send an email to sales@koretech.com.

About Kore Technologies

Kore Technologies is a leading provider of enterprise integration, business intelligence and eCommerce Web solutions for MultiValue and Microsoft SQL Server databases. Kore specializes in providing business-to-business connectivity and best-in-class product integration with its flagship product--Kourier Integrator. Leveraging this technology and Kore's extensive experience with enterprise applications and database systems, Kore delivers solutions for integrating disparate applications, building data warehouses, developing business intelligence systems, and creating integrated eCommerce applications. For more information, visit www.koretech.com.

E-COMMERCE MADE E-Z

Continued from page 11

```
O4WPaypal("ADD", "addrtn",
"paypalmgr@revelation.com",
"sample item", "samp001", "",
"10.99")
```

generates the HTML in figure 3 and, when clicked, generates the action in figure 4.

Finally, the "View Cart" action accepts a subset of those name/value pairs, specifically "cmd" and "business", and adds "display", whose value should be set to "1". The O4W View Cart PayPal request:

```
O4WPaypal("CART", "cartrtn",
"paypalmgr@revelation.com")
```

generates the HTML in figure 5 and, after a few more items were added to the cart, the action in figure 6 was generated.

By using these three Website Payments Standard APIs, you can add the ability to immediately accept credit cards or cash from any customer without developing your own custom software solution. If the "bottom line" in your business is to keep a healthy bottom line, the ability to get paid quickly and easily has to be a top priority, and PayPal has made that the easy part of e-commerce. **IS**

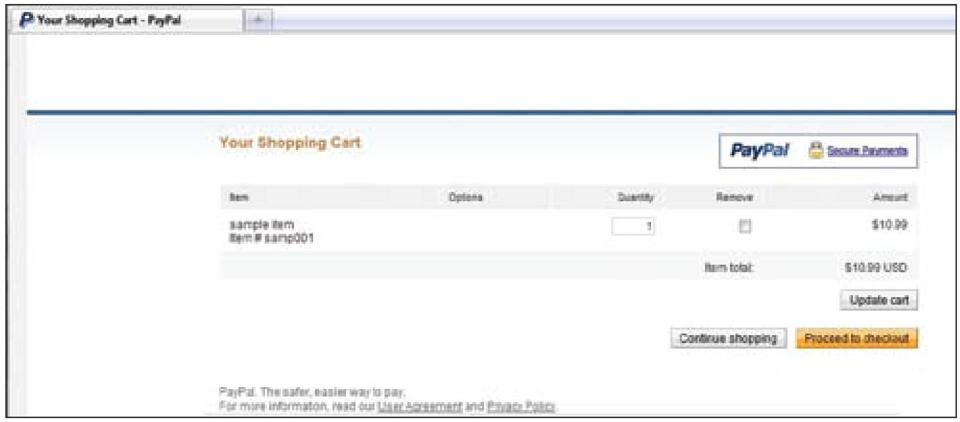


Fig. 4

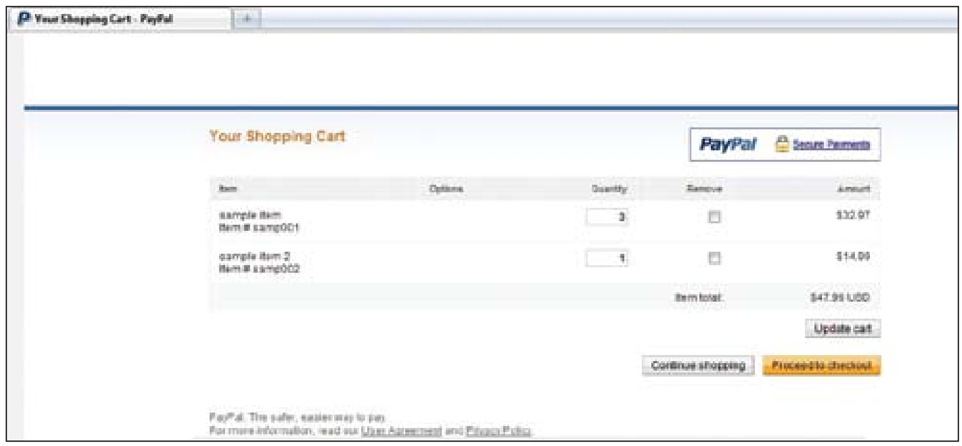


Fig. 6

```
<form enctype="multipart/form-data" method="post" action="https://www.paypal.com/cgi-bin/
      → webscr" name="o4wpostform">
   <input name="cmd" value="_cart" type="hidden">
   <input name="business" value="paypalmgr@revelation.com" type="hidden">
   <input name="currency_code" value="USD" type="hidden">
   <input name="lc" value="US" type="hidden">
   <input name="add" value="1" type="hidden">
   <input name="item_name" value="sample item" type="hidden">
   <input name="amount" value="10.99" type="hidden">
   <input name="button_subtype" value="products" type="hidden">
   <input name="bn" value="PP-ShopCartBF:btn_cart_SM.gif:NonHosted" type="hidden">
   <input name="item_number" value="samp001" type="hidden">
   <input name="shopping_url"
value="http://www.revelation.com/o4w/oecgi3.exe/O4W_ORDERFORM?O4WEvent=Paypal&amp;
      → O4WControlBackup=addrtn" type="hidden">
</form>
```

Fig. 3

```
<form enctype="multipart/form-data" method="post" action="https://www.paypal.com/cgi-bin/
            → webscr" name="o4wpostform">
   <input name="cmd" value="_cart" type="hidden">
   <input name="business" value="paypalmgr@revelation.com" type="hidden">
   <input name="display" value="1" type="hidden">
   <input name="shopping_url"
value="http://www.revelation.com/o4w/oecgi3.exe/O4W_ORDERFORM?O4WEvent=Paypal&amp;
            → O4WControlBackup=cartrtn" type="hidden">
</form>
```

Fig. 5

MULTIVALUE AND QUICKBOOKS, PART 1: IMPORTING WITH IIF FILES.

Continued from page 14

definitions within a single IIF file and QB will handle them accordingly.

Thus, starting with the first row, you will define that first section of data. Assuming nothing, you would export everything that is pertinent to the managing of these AR records, meaning the customer info and the specific AR transaction.

QB has tables called CUST, INVITEM and TRNS. CUST is the obvious Customer file, INVITEM is similar to a Product file and TRNS is for Transactions. TRNS files have additional definitions depending on what kind of transaction they are: INVOICE, PAYABLE, PAYROLL etc. We will be using INVOICE.

QB documents all the fields pertinent to each table. There are plenty of fields per table and some are required. Besides the obvious, there is no indication of being required.

A QB Invoice transaction requires something to be "sold." A typical MultiValue A/R record would only contain the customer, dates and dollars, and the primary key may refer back to the Order or Production system for the contents of the sale. QB does not need to know how many wingnuts were sold. But "something" needs to be sold as a line item in QB as the INVOICE table looks to its lines for the actual data.

The INVITEM file would be used sparingly to represent the kinds of A/R imports, typically sales and perhaps credits. This entire function would be subordinate to the regular MultiValue Invoice/Credit updates. Thus, you would have one or two INVITEMs, perhaps called INVOICE and CREDIT.

A blessing with importing IIF files into QB is if the record already exists, you won't get a warning message nor will it overwrite. Thus the second and consequent sessions will receive these IN-

VITEMs but not complain if they are already there. Likewise for the CUST file.

Coming from a MultiValue mentality, QB does have an issue. It uses the CUST.NAME field as its unique primary key. This runs contrary to a separate customer number/code. Thus, in both cases, the CUST.NAME field exported is “123456-Chicago Gasket Company”. QB does have a separate Name and Address area but many of the customer views simply show this CUST.NAME field so the code alone is not helpful nor is the name that guaranteed to be unique.

Sidebar — should the customer name change on the MultiValue system, you could end up with two customer records in QB for the same account. QB does have a ‘merge customer’ facility that easily fixes this. It is probably easier to import the redundant customer record instead of maintaining what you sent to QB from MultiValue. The most recent name is probably the one they will merge into anyway.

For brevity, I will show the left-hand portion of an IIF file that exports an INVITEM, a CUST record and an invoice TRANSaction (fig. 1). It is classic CSV format with spaces added for clarity. I tend to use double quotes for all CSV rows whether necessary or not.

Lines that start with an exclamation mark define the format of those records starting with the table name. !TRNS defines the TRNS table fields being imported. TRNS is the data.

Line 1 defines the INVITEM records. Line 2 is a data record, line 3 ends the INVITEMS.

Line 4 defines the CUST records, lines 5, 6 & 7 are data records, line 8 ends the CUST.

Line 9 defines the generic TRNS rows with the TRNSTYPE column.

Line 10 defines each data line in each transaction. It starts out looking similar to the !TRNS row but it contains the quantity of each of the INVITEMs “sold.” For regular A/R usage, it could be just one line stating “Invoiced Sale”. You could replicate the entire contents of the existing sales order.

Line 11 ends the TRANS header.

Line 12 matches the definitions on line 9. Line 13 matches the definitions on line 10.

Line 14 ends the TRANS detail row.

If you note, this IIF example has the INVITEM defined first, then the inventory item itself. Then the CUST table is defined followed by all the customers. Finally the TRANS table is defined then followed by all the transactions.

You could put all of the definitions in the first few rows. Then the following rows would be the data. Each data row follows the definition of the first cell in each row. This sequence would more logically match the READNEXT flow of processing MultiValue records instead of accumulating the INVITEM, CUST and TRANS tables separately to append to the larger IIF file.

When importing, use test companies in QB. Otherwise you will have to go through the existing company and delete any errant records. This cannot be emphasized enough.

While we may not be consulted about the company’s decision to move the Accounting from MultiValue to QB, we should embrace the opportunity to bridge between the two. As mentioned earlier, accounting software borders on being a commodity. The real intellectual portion of the company’s system is in the production and sales/order modules which will probably remain as MultiValue for quite a long time. **IS**

MARK JOHNSON is an independent MV consultant with over 30 years experience on virtually all platforms. He specializes in maintaining legacy systems, helping his clients maximize their investment while participating with current technologies. When the time is right, he facilitates migrations to the proper contemporary platform. He can be reached at majprogramming@gmail.com or 908-601-4052.

```
001 !INVITEM, NAME, INVITEMTYPE, DESC, ACCNT, PRICE
002 INVITEM, CHARGES, SERV, Invoiced Sale, 100.00
003 ENDINVITEM
004 !CUST, NAME, BADDR1, BPHONE1, COMPANYNAME
005 CUST,”1234-GASKET CO”, “123 MAIN ST”, “555-5555”, “GASKET CO”
006 CUST,”4545-JONES CO”,”456 ELM ST”,”666-6666”,”JONES CO”
007 CUST,”1111-SMITH CO”,”999 OAK ST”,”888-8888”,”SMITH CO”
008 ENDCUST
009 !TRNS, TRNSTYPE,DATE,ACCNT, NAME, AMOUNT, DOCNUM
010 !SPL, TRNSTYPE, DATE, ACCNT, NAME, AMOUNT, DOCNUM
011 “ENDTRNS”
012 “TRNS”,”INVOICE”,”04/01/2010”,”AR”,”1234-GASKET CO” “, “200.00”, “B123”
013 “SPL”,”INVOICE”,”04/01/2010”,”Sales Income”,””,”-200”,”B123”
014 ENDTRNS
```

Fig. 1

B Y N A T H A N R E C T O R

Sending E-mail from your MultiValue Programs

Part 1

E-mail is part of every day business life. We receive it and send it on a daily basis. Our vendors and customers send it on a daily basis. It is amazing that many companies still don't have e-mail sending and receiving integrated directly into their business systems.

Many times, emails are sent using Windows GUI programs tied into the MultiValue systems, or through the terminal emulators. This works well when you are actively trying to send an e-mail from a process, but not when you are updating your business processes in the background or need to send status information to customers or employees when process are done without bothering the person entering the information.

The most powerful thing about the MultiValue business environments is the enhancements scripting and programs you have built into the database. Many other database systems out there require most of your business logic to reside on a separate server, or outside the database, which means when something in the database is updated, programs have to remember to run other processes as part of that update.

Since the business rules, logic, and other business processes run on our MultiValue database, we don't have to remember to do things, nor do we have to worry about creating a Multi-Tier programming environment, or problems with running the same process from on different computers.

The sender of the e-mail isn't always human, either. But most e-mails need to be readable by humans.

Integrating e-mail sending functions into your business application is actually really easy. The hard part is choosing which e-mail server you are going to use to send your e-mail.

In this article we will talk about how to send emails from different types of e-mail systems and the pro and cons of using those systems.

What is e-mail and how do we use it?

Ok, so this can sound like a dumb question. We all know what e-mail is and how to use it. E-mail allows us to communicate information between two or more parties. What many people forget is that the "parties" part of that statement isn't necessarily a human person. It could be a business application, a blog, a newsgroup, or a status system.

The sender of the e-mail isn't always human, either. But most e-mails need to be readable by humans. An example of this is bounced e-mail messages. We all have seen these. When we open them in our e-mail client, we see a message about why it bounced and maybe a part of the original e-mail. There is more to the e-mail than this though. There is computer parsable information that tells send servers, programs, or bots why this e-mail bounced in a status or error code, where it came from, which server it bounced from, in addition to a human readable description of the bounce reason.

I'm not going to go to in-depth in this article on bounce messages and no human readable e-mails. I just want to make you aware of it.

E-mail Format

Before we get into how to actually send the e-mail, we have to understand how to create the actual e-mail. At a very high level, the format of e-mail is very simple. It can get very complex if you are trying to implement a full mail server with all the features for mail lists and spam filters, but for our purpose we will look only at the basics. It will likely be all you need, especially if you will only be sending e-mails.

An e-mail consists of two major sections: header and body. The body is the easy part. It is the information that you want the receiver to read. There is really only one requirement for the body section. A blank line MUST exists between the last line of the header section and the first line of the body section. Other than that, there is no real requirement except for how to format the information you want the user to read.

I know that it may not make complete sense, especially when you get into attachments. But let me get through how to create the header section first, and then things may make sense.

The header section of the e-mail provides the key information that needed in order for e-mail to be received successfully by its destination. There are only four required header fields, but others are recommended.

The required headers fields are "From", "To", "Date", and "Subject."

"From:" — This header is supposed to contain the e-mail address and name of the person or process sending the e-mail. The reality is that this can contain anything you want as long as it looks like an e-mail address.

"To:" — This header lists the e-mail addresses and names of the people or processes you want to receive this e-mail. Again, like the "From:" header, this really can contain anything you want as long as it looks like an e-mail address. Depending on the routines you use to actually send the e-mail, this header is used by the sending program or ignored. When I get into the actually sending part I'll talk about that more.

"Date:" — This header contains a date and time of when the e-mail was created. This date and time needs to be formatted in a specific way. This format is called RFC 1123. It basically consists of the following: Day-Of-Week, Day MonthName Year 24Hour:Min:Sec Time-Zone

`Ie: Wed, 9 Sep 2008 13:45:00 EST`

"Subject:" — The subject of the e-mail. Pretty straight forward.

Each header field must have a ":" and a space or tab between the header field and the data, followed by a CRLF to terminate the line. For those that don't know the values of CRLF, it is defined as a Carrage Return and Lind Feed, which is CRLF = CHAR(13) : CHAR(10) in MultiValue Basic.

There are many other headers, but I'm going to focus on a simple e-mail first before I get into the more complex headers.

In figure 1, I've included a sample of a simple e-mail with the headers and a text body.

Because of the origins of e-mail, e-mail clients will assume the body section of

Continues on page 29

```
To: emailexample@intl-spectrum.com

From: nathan@intl-spectrum.com

Date: Weds, 9 Sep 2008 13:45:00 EST

Subject: Simple Email

hello, this is a simple email message.
```

Fig. 1

Formatting Queries Without A Dictionary Descriptor in UniVerse

Log onto any MultiValue application that has been in existence for three years, find a commonly used file, say CUST, and list it's dictionary. What do you see? Definitions for LNAME, LAST.NAME, L.NAME, LST.NAME, LNAME40, etc. Upon closer examination you discover that field (attribute) is also defined as X, XX, PGM030.LNAME.RPT, and on and on. Log on to an application that has been in use for 30 years and you could find several hundred dictionary entries for a file with only 20 fields.

Obviously what has happened is that if LNAME is originally defined as 15 characters wide and a report wants to use it in a column 13 characters wide, there you go. Another dictionary entry. Then in an ad hoc query, someone wants the data in a column 20 characters wide. Plop. There's another "temporary" definition that never gets cleaned up.

In UniVerse there is a solution to this problem. All of the formatting things you can do from a dictionary descriptor you can do from the command line, or from within a Paragraph, Proc, EXECUTE, etc.

Staying with column width as an example, it's quite simple. In your query statement, after naming the column you want displayed, simply add the justification and width specification with the FMT keyword. For example,

```
LIST CUST LNAME FMT "35L"
```

Conversion codes are simply specified with the CONV keyword.

```
LIST CUST LNAME FMT "35L" BAL.DUE CONV "MD24,$"
```

Column headings, anyone? The key word is COL.HDG. Also notice that the column heading text can include the 'L' option to break the heading into multiple lines.

```
LIST CUST LNAME FMT "35L" BAL.DUE CONV "MD2,$" COL.HDG "Amt'L'Owed"
```

But what if you want to perform a calculation or some kind of string manipulation? Then you have to use an I-descriptor, don't you? As you might suspect, the answer is no. Anything you can code in the expression of an I-descriptor can be done on the command line using the EVAL keyword. Let's combine the last name and the first initial to give us the names of the customers listed.

LIST CUST EVAL "LNAME:', ':FNAME[1,1]"

While these are useful for ad hoc queries, these keywords can also be used to write permanent reports that are relatively immune to someone changing a descriptor and having it affect your report, like making it switch to non-columnar format, for example. Remembering that a line can end in an underscore character to indicate "continued on the next line," you might write a Paragraph like this.

```
PA
SORT CUST WITH BAL.DUE > "0" _
EVAL "LNAME:', ':FNAME[1,1]" FMT "30L" COL.HDG "Name" _
BAL.DUE CONV "MD2,$" FMT "10R" COL.HDG "Amt Owed" _
HEADING "Customer Owing Us Money" _
LPTR
```

For additional command line formatting abilities, see chapter 3 of the UniVerse Guide to Retrieve manual. IS

Do you have a Tech Tip to share? E-mail it to editor@intl-spectrum.com

SENDING E-MAIL FROM YOUR MULTIVALUE PROGRAMS

Continued from page 27

your e-mail is plain text. To tell a client what kind of information is in the body section, you need to include a header called Context-Type.

There are a lot of different context types out there. For most e-mails you will only care about four context types: text/plain, text/html, multipart/alternative, and multipart/mixed.

For plain text e-mails, you would want to add the "context-text: text/plain". There are a few other headers that I recommend you include to help you get your e-mail safely past spam filters. These headers are MIME-Version, X-Mailer, and Content-Transfer-Encoding.

MIME-Version will always have a value of: "MIME-Version: 1.0". X-Mailer header is a free form text description of where this e-mail comes from. Many times the e-mail sending routine you use will add this, but not always. There are many e-mail filters that specifically look to see if this header exists. If it doesn't have a X-Mailer header, then it flags it as a higher possibility of spam.

Context-Transfer-Encoding is pretty straight forward as well. Unless you want to get into encoding the text of your e-mail, you should set it's value as "8bit". If you know you won't have any characters above CHAR(128), then you can use "7bit". Most text-only e-mails won't have characters above CHAR(128) since that is in the unprintable range.

Take a look at figure 2 to see an e-mail with all the headers included.

There is one last thing to keep in mind when creating a e-mail — most e-mail servers impose limits of 1,000 characters per line of e-mail, including the CRLF. For readability, it is suggested that you format your e-mails to a maximum of 80 characters per line.

Please keep in mind that I've only scratched the surface of creating e-mails. There is much more. In fact there are whole documents, web sites, and books that explain just how to create the e-mail headers and body. I will get into more of this later, but I wanted to give you something to start with. You can find out more about e-mail protocol and formats in the book Internet E-mail Protocols: a Developer's Guide by Kevin Johnson, or you can just do a search on the Internet.

In the next article, I'll cover how to use the information in this article to send an e-mail. **IS**

```
To: emailexample@intl-spectrum.com
From: nathan@intl-spectrum.com
Date: Weds, 9 Sep 2008 13:45:00 EST
Subject: Simple Email
Content-Type: text/plain
Content-Transfer-Encoding: 7bit
X-Mailer: International Spectrum Email Sending 1.0
MIME-Version: 1.0

hello, this is a simple email message.
```

Fig. 2

CLIF NOTES: I RESOLVE TO NOT WRITE ANY NEW YEAR'S RESOLUTION COLUMN...NEXT YEAR

Continued from page 30

- Experiment with several different IDEs.

If you are still writing MultiValue Basic code by using the ED editor, you are wasting way too much of your valuable time at the keyboard. There are a variety of source code writing tools available that work well in developing MultiValue code. Whether it is a full-blown IDE specifically targeted towards MultiValue or simply a GUI-based editor, getting away from a line-at-a-time editor is one of the fastest and cheapest ways you can achieve a major productivity boost almost overnight. Shucks, even Windows Notepad or the *nix VI editor are better than flogging it out with ED!

Okay, okay. Go ahead and keep using ED for dictionary descriptors.

- Complete at least one online tutorial each week.

If your MultiValue vendor provides online tutorials, it's a shame not to take advantage of them. But the topic you select does not necessarily have to be MultiValue. With just a couple minutes of Web search, you can find tutorials on just about any topic. Interested in expanding your abilities in Web related programming? PHP, JavaScript, Ajax — there are numerous tutorials available at no charge on the Web. Interested in learning or becoming better in SQL? I just searched on the words "SQL tutorial" using Google and got 627,000 results.

Those are just a few suggestions to get your ideal factory going. Obviously, there could be many, many more. Come up with one or more of your own that speak to your own personal goals and interests. Whether 2011 turns out to be a good year or not is largely dependent on what we decide to do with it.

I'll see you at the 2011 International Spectrum Conference! **IS**

CLIF NOTES: I RESOLVE TO NOT WRITE ANY NEW YEAR'S RESOLUTION COLUMN...NEXT YEAR

Continued from page 31

ize your existing MultiValue application into something that management doesn't view as an obsolete technology to be dumped.

- Learn at least two ways to connect from your MultiValue system to a non-MultiValue system and extract and update data.

Except in the smaller shops, it is getting rare to see a MultiValue system as the only computing platform in the company. Most of the MultiValue systems I work with exist side-by-side with *nix servers, Windows servers, and Web servers running a variety of non-MultiValue databases. The demand is increasing to tie these various data repositories together in order to provide a complete view of the enterprise's information resources. Business Intelligence is an important part of modern business, and the modern MultiValue system must be a part of it.

- Read or reread your MultiValue platform's entire documentation set.

One of the problems with working with a particular platform for more than just a couple of years is that you fall into the habit of just continuing to do things the way that you have always done them. But all of the MultiValue vendors are continuously adding additional capabilities to their products. Especially if the platform you are currently working on is not the same one that you began your MultiValue career on, you might be amazed at all the different ways of doing things and some of the abilities you now have available to you. Even if you have been on the same MultiValue platform for several years, I can almost guarantee that if you reread the documentation, you will find a number of things that they quietly slipped in during the evolution of the product that you didn't know about or have simply forgotten.

Continues on page 29

I Resolve To Not Write Any New Year's Resolution Column...Next Year

Clif Notes

B Y C L I F T O N O L I V E R

I think I set a new efficiency record this New Year. I have never been into New Year's resolutions very much. With 365 days in a year, I don't understand waiting for a particular one in order to make a decision to accomplish something in the next 364. But, it seems to be a cultural tradition, not unlike the ritual of eating certain foods on New Year's Day in a magical attempt to woo Lady Luck or the Goddess Ifni to smile upon us. Nevertheless, this year I decided to participate.

Over the years, I have observed that most resolutions have a common characteristic. They are made to be broken. According to some reports, the most common New Year's resolution is to lose weight and get in shape. This one is normally broken within 12 hours, typically at the New Year's Day feast. That's not bad from an efficiency standpoint. But I thought I could do better.

I resolved not to make any New Year's resolutions.

There you have it. Resolution made and simultaneously broken. Duration: zero seconds.

Of course, joking aside, the entire process can be seen in a more positive light. New Year is merely a convenient trigger event to remind us to pause, reflect, and adjust our short term goals in order to keep moving towards our long term goals. This is a Good Thing. So in that spirit, I would like to offer a few possible resolutions for MultiValue developers who tend to get into their comfort zone and allow opportunities to increase the longevity of their beloved MultiValue applications or enhance their own career opportunities to pass them by. I have referred to this in previous columns as slipping into becoming a MultiValue Couch Potato. (None of us are completely immune, including me.)

Here are a few of my suggested resolutions for losing the brain flab and tightening the thinking muscles in 2011.

- Learn a new programming language.

MultiValue Basic is an excellent and flexible language. But now that many of the platforms are running on an underlying host operating system (Unix, Linux, Windows), a knowledge of the languages used in those environments can really add to what you are able to do in your applications.

- Learn a GUI-based language or tool that will let you put a modern user interface on your MultiValue applications.

The green screen is dead. Whenever I talk about this at a conference or user group meeting, there always seems to be somebody who wants to get into an argument about how much more efficient green screen is for data entry, how happy their users are with the way things are, etc. Well, that's not what I observe and hear from users. The world expects GUI. If you don't give it to them, I am willing to bet money that there is a movement, not only among your users but also in IT management, to kick the MultiValue system to the curb (and possibly you along with it).

- Pick the ugliest green screen program in your application and rewrite it using a GUI-based tool.

Once you learn how to write a GUI-based program that can talk to your brand MultiValue database, do an experiment and gather some statistics. Take that really horrific cluttered green screen program that everybody complains about and rewrite it with a GUI. Keep accurate records of how long each step in the process takes. You now have a more accurate way to estimate the time and money it would take to modern-

Continues on page 30